

İçindekiler

SQL

1. BÖLÜM

Veri Sorgulama (Quering Data)	11
Veri Sorgulama (Select)	13
Sorgu Sonucunu Sıralama (Order By)	15
Yinelenen Satırları Kaldırma (Distinct)	16

2. BÖLÜM

Veri Filtreleme (Filtering Data)	17
Where	19
Limit	21
In	22
Between	24
Like - ilike	25

3. BÖLÜM

Tabloların Birleştirilmesi (Joining Multiple Tables)	27
---	-----------

4. BÖLÜM

Veri Gruplama (Grouping Data)	33
Group by	35
Having	36
Having vs. Where	36

5. BÖLÜM

Set Operations	39
Union	41
Intersect	44
Except	44

6. BÖLÜM

Alt Sorgu (Subquery)	45
Alt Sorgular	47
Any Operatörü	51

7. BÖLÜM

Veri Ekleme Güncelleme ve Silme işlemleri (Modifying Data)	53
Insert	55
Update	58
Update Join	60
Delete.....	61

8. BÖLÜM

Transaction.....	63
Begin	65
Commit.....	66
Rollback.....	67

9. BÖLÜM

CSV İmport Export	69
--------------------------------	-----------

10. BÖLÜM

Tablo Yönetim (Managing Tables)	77
Veri Türleri.....	79
Tablo Oluşturma.....	80
Constraints	80
Create Tables AS	82
Alter, Rename, Drop, Truncate Table.....	83
Temporary Table.....	92
Copy Table.....	93

11. BÖLÜM

Kısıtlamalar (Constraints).....	95
Primary Key	97
Foreign Key.....	98
Check Constraint	98
Unique Constraint	99
Not-Null Constraint	100

12. BÖLÜM

Matematik İşlemleri Tarih Zaman Fonksiyonları.....	101
(Mathematical Operations – Date Time Functions)	

13. BÖLÜM

psql Komutları	109
-----------------------------	------------

14. BÖLÜM

Örnek Yöntemler (SQL Recipes).....	115
---	------------

PLpgSQL

1. BÖLÜM

PLpgSQL'e Giriş	123
String Sabitleri	125
Block Yapısı	126

2. BÖLÜM

Değişkenler ve Sabitler (Variables & Constants)	129
Değişkenler	131
Veri Türlerini Kopyalama	132
Subblock	133
PLpgSQL Satır Türleri	134
Kayıt Türü	135
Sabitler	136

3. BÖLÜM

Mesajlar ve Hatalar (Reporting Messages and Errors)	137
--	------------

4. BÖLÜM

Kontrol Yapıları (Control Structures)	141
If Then	143
Else-If	144
If-Then-Elseif	145
Case	150
Loop	154
While Loop	154
For Loop	157
Sonuç Kümesi Üzerinde (Query Result) Loop Döngüsü	158
Dinamik Bir Sorgunun Sonuç Kümesinde Loop Döngüsü	159
Exit İfadesi	160
Continue İfadesi	162

5. BÖLÜM

Veri Kullanıcı Tanımlı Fonksiyonlar (User - Defined Functions)	163
Fonksiyon Oluşturma	165
Fonksiyonları Çağırma	168
Fonksiyon Parametreleri	169
Fonksiyon Overloading	172
Tablo Döndüren Fonksiyon	174
Fonksiyon Silmek	176

6. BÖLÜM

Stored Procedures	177
--------------------------------	------------

	7. BÖLÜM	
Triggers		187
	8. BÖLÜM	
Views		197
	9. BÖLÜM	
Indexes.....		203
B-tree Index.....		206
Hash Index.....		206
Brin Index		207

SQL

1

Veri Sorgulama (Querying Data)

- Veri Sorgulama (Select)
- Sorgu Sonucunu Sıralama (Order By)
- Yinelenen Satırları Kaldırma (Distinct)

SELECT

Tablolardan ve view'lerden select cümleleri ile veri sorgulaması yapılır.

ÖRNEKLER:

```
SELECT kolon1, kolon2, kolon3 FROM tablo_adı;

# Bir tablodaki tüm kolonları sorgular.
SELECT * FROM rental ;

# Belirli kolonları sorgular.
SELECT country_name FROM adress ;
--veya
SELECT first_name, email, username FROM staff ;

# Spesifik bir kolon belirtmeyip '*' ifadesi kullandığınız
takdirde belirttiğiniz tablodaki tüm satır ve sütun sonuçları
gelecektir.
Select * from rental;

# Tabloda birden çok kolondaki değerleri çekmek istediğinizde
',' ile tek tek kolon adı belirtmeniz durumunda, belirttiğiniz
sıra ile sorgu sonucu gelecektir.

SELECT address_id FROM address ;
SELECT country_id FROM city ;
SELECT first_name, email, username FROM staff ;
```

Kolon Birleştirme

Sql cümlesi first_name ve last_name satırlarının değerlerini tek bir kolonda getirir, diğer satırda da email kolonu yer alır.

```
SELECT first_name || ' ' || last_name, email FROM customer;
```

Alias

Sorgu ile kolona takma ad verilir.

```
SELECT kolon_adı AS alias_name FROM tablo_adı;

# Aşağıdaki sorgu first_name ve last_name satırlarını tek bir
kolonda bir araya getirip, kolona 'name' takma adını vermek
için kullanılır.

SELECT first_name || ' ' || last_name as name , email FROM
customer;

SELECT 20/5 as sonuc ;
SELECT last_name as sure_name , first_name from customer ;
SELECT first_name, last_name surname FROM customer;

# Sütunun takma adı boşluk içeriyorsa çift tırnak içerisinde
takma ad yazılır.

SELECT first_name || ' ' || last_name "full name" FROM customer;
```

Sorgu Sonucunu Belirli Bir Sıraya Göre Listeleme

```
SELECT select_listesi FROM tablo_adı ORDER BY sıralama_ifadesi1  
[ASC | DESC], ..,sıralama_ifadesiN [ASC | DESC];  
  
SELECT actor_id, last_name FROM actor ORDER BY actor_id DESC;  
  
SELECT title, fulltext FROM film ORDER BY title ASC;  
  
SELECT first_name, last_name FROM customer ORDER BY last_name  
DESC, first_name ASC;
```

Bir tablodan veri sorguladığınızda, `SELECT` ifadesi satırları düzensiz bir sırada döndürür. Sonuç kümesinin satırlarını sıralamak için `SELECT` cümlesinde `ORDER BY` ifadesi kullanılır. `ORDER BY` ifadesi, bir `SELECT` cümlesi tarafından döndürülen satırları, bir sıralama ifadesine göre artan veya azalan düzende sıralamanıza olanak sağlar.

Sonuç kümesini birden çok sütuna veya ifadeye göre sıralamak istiyorsanız, bunları ayırmak için iki sütun veya ifade arasına virgül (,) koyulması gerekir. Satırları artan düzende sıralamak için 'ASC' seçeneği ve azalan düzende sıralamak için 'DESC' seçeneği kullanılır. ASC veya DESC seçeneklerinden biri belirtilmezse, `ORDER BY` varsayılan olarak ASC'yi kullanır.

```
SELECT first_name, LENGTH(first_name) len FROM customer ORDER BY  
len DESC;
```

Sorgu customer tablosunda first_name sütunundaki değerlerin LENGTH () fonksiyonu ile (harf olarak) uzunluğunu hesaplar.

Uzunluk sonucunun kolonuna alias olarak len yazar ve bu kolona göre azalan değerde sıralama yapıp sonuçları getirir.

Sonuç Kümesinden Yinelenen Satırları Kaldırmak

```
SELECT DISTINCT kolon1 FROM tablo_adı;  
  
SELECT DISTINCT kolon1,kolon2 FROM tablo_adı;  
  
SELECT DISTINCT last_update FROM country ;  
  
SELECT DISTINCT last_update, country FROM country order by  
country desc ;
```

DISTINCT ifadesi, bir sonuç kümesinden yinelenen satırları kaldırmak için SELECT cümlesinde kullanılır. DISTINCT ifadesi, yinelenen her grup için bir satır tutar. SELECT cümlesinin seçim listesindeki bir veya daha fazla sütuna uygulanabilir.

Birden çok sütun belirtirseniz, DISTINCT ifadesi bu sütunların değerlerinin birleşimine göre kopyayı değerlendirir.

2

Veri Filtreleme

(Filtering Data)

- Where
- Limit
- In
- Between
- Like - ilike

İçerik :

SQL

- 1- Veri Sorgulama / 4 syf
- 2- Veri Filtreleme / 8 syf
- 3- Tabloların Birleştirilmesi / 4 syf
- 4- Veri Gruplama / 3syf
- 5- Set Operations / 4 syf
- 6- Alt Sorgu (Subquery) / 6 syf
- 7- Veri Ekleme Güncelleme ve Silme işlemleri / 7 syf
- 8- Transaction / 3 syf
- 9- CSV Import Export / 5 syf
- 10- Tablo Yönetimi / 16 syf
- 11- Kısıtlamalar / 4 syf
- 12- Matematik İşlemleri Tarih Zaman Fonksiyonları / 5 syf
- 13- PSQL Komutları / 4 syf
- 14- Örnek Yöntemler / 3 syf

PLpgSQL

- 1- PLpgSQL'e Giriş / 3 syf
- 2- Değişkenler ve Sabitler / 6 syf
- 3- Mesajlar ve Hatalar / 2 syf
- 4- Kontrol Yapıları / 20 syf
- 5- Kullanıcı Tanımlı Fonksiyonlar / 12 syf
- 6- Stored Procedures / 7 syf
- 7- Trigger / 7 syf
- 8- View / 3 syf
- 9- Index / 3 syf

Meryem Sağol

Sakarya Üniversitesi Endüstri Mühendisliği mezunudur. Veritabanı alanında uluslararası geçerliliğe sahip bir çok sertifikaya sahiptir.

2019'da bilişim sektöründe çalışmaya başlamıştır.

Dataera'da veritabanı yöneticisi olarak çalışmaktadır.

